

The control bit in position 8 essentially offsets address mapping by 128 (decimal) or 80 (hex). The 'access address' (address + control bit) to access DD RAM is:

DISPLAY POSITION		1	2	3	4	5	6	7		79	80
ACCESS ADDRESS	Decimal	128	129	130	131	132	133	134		206	207
ACCESS ADDRESS	Hex	80	81	82	83	84	85	86		CE	CF

The above table requires two refinements to cover multiple line displays. Unlike CRT displays, when the cursor reached the end of the first line of a *two*-line display, it does *not* automatically proceed to line 2. The cursor must be moved to line 2 before characters can be displayed there. The DD RAM address for all two-line displays, regardless of size, begins at access address 180 decimal (C0 hex). For example, a 2 x 20 display has line 1 addresses of 128 to 147 (decimal) and line 2 addresses of 180 to 199.

Four-line displays (16 and 20 characters per line) follow the operation for two-line displays with one complication. Unlike two-line displays, line 1 *does* wrap around, but to *line 3*. For example, with a 4 x 20 display and the cursor at the end of line 1, the 21st character will display at position 1 on line 3. Similarly, line 2 wraps around to line 4. The transition from line 1 to line 2 as well as lines 2 to 3 and 3 to 4 are the same as for two-line displays. These line-to-line transitions must be handled by repositioning the cursor to the proper DD RAM address. The following three table contain the access address maps for one, two, and four-line displays.

ONE-LINE DISPLAY

DISPLAY POSITION			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...	79	80
LINE 1	Access	Dec	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	...	206	207
	Address	Hex	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	...	CE	CF

TWO-LINE DISPLAY

DISPLAY POSITION		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...	39	40	
LINE 1	Access Address	Dec	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	...	167	168
		Hex	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	...	A6	A7
LINE 2	Access Address	Dec	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	...	230	231
		Hex	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	...	E6	E7

FOUR-LINE DISPLAY

DISPLAY POSITION		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		19	20	
LINE 1	Access Address	Dec	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142		146	147
		Hex	80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E		92	93
LINE 2	Access Address	Dec	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206		210	211
		Hex	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE		D2	D3
LINE 3	Access Address	Dec	148	149	150	151	152	153	154	155	156	157	158	159	160	161	162		166	167
		Hex	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F	A0	A1	A2		A6	A7
LINE 4	Access Address	Dec	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226		230	231
		Hex	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF	E0	E1	E2		E6	E7

For 4 x 16 LCD displays, refer to manufacturer's specifications for map of Display Data Address RAM

SETTING CHARACTER ENTRY POSITION [STEP 4 Cont'd]: To display information, set the initial cursor position prior to outputting information. The INITLCD command defaults to the first position on the display. A second BASIC command, OUTLCD, is used to set cursor addresses as well as execute other functional instructions. The command format is: **OUTLCD <access address>**

Examples are:

OUTLCD 128	'Establishes position 1 (line 1) as character entry address
OUTLCD \$80	'Same as above, except address is in hex notation
OUTLCD 192	'Sets position 1, line 2 as the character entry address
OUTLCD 148	'Sets character entry position at mid-point of a 1x40 display
OUTLCD 143	'Sets character entry position at end of line 1 of 1x16 display
OUTLCD 212	'Sets character entry position to #1, line 4, of a 4x20 display

WRITING TO THE DISPLAY [STEP 5]: BASIC's PRINT command is used to write text and data to a display. MultiTRAX BASIC expands the capability of the PRINT command through a device driver, DEVICE 6, to minimize program code. For example:

10	Device 6	Semi-colon at end of PRINT line to suppress line feed & carriage return
20	PRINT "Testing Display";	

PROGRAM EXAMPLES: Here are example programs, including initialization, configuration and data entry. The first example contains fundamental code to place and display text. Change the first argument in line 10 to match your display; refer to 'Summary of Initialization & Configuration' on page 5. Example [LCD50]:

10	INITLCD 0,2,4	'1-Line, 5x7 dot font, begin pos #1; entry like CRT screen
20	DEVICE 6	'Call the LCD display driver
30	OUTLCD 128	code; example to set entry point to pos #1
40	PRINT "Display Test";	'Prints text to display

The next example illustrates how to display both text and data. The text 'Temperature' is positioned at the left side of the display. The entry point is changed to display three simulated temperature readings in an update fashion while maintaining the temperature label on the display. Example [LCD51]

10	INTEGER TEMP1,TEMP2,TEMP3	'Declare three variables
20	TEMP1=40:TEMP2=55:TEMP3=125	'For this example, define variable values
30	INITLCD 0,2,4	
40	DEVICE 6	
50	PRINT "TEMPERATURE: ";	'Print label on display
60	OUTLCD 142	'Set entry position to position #15
70	PRINT TEMP1;	'Print value TEMP1 beginning pos #15
80	SUSPEND 200	'Wait 2 seconds then ...
90	OUTLCD 142	'Set entry position (back) to pos #15
100	PRINT TEMP2;	'Overwrite TEMP1 value with TEMP2
110	SUSPEND 200	
120	OUTLCD 142	
130	PRINT TEMP3;	'Overwrite again
140	STOP	

This example illustrates a technique to provide lengthy instructions by scrolling information on a display in ticker tape fashion. Example [LCD52]

10	STRING A\$(120)
20	STRING B\$
30	INTEGER I,J,N
40	INITLCD 0,3,4

```

50 OUTLCD 167
60 DEVICE 6
70 A$="A TEST OF THE MultiTRAX IS ABOUT TO BEGIN .... TURN OFF SENSOR #1"
100 FOR I=1 TO 65
110 B$=MID$(A$,I,1)
120 PRINT B$;
130 SUSPEND 20
140 NEXT I

```

POINTS TO REMEMBER: Characters do NOT automatically flow from line 1 to 2. Characters may shift off the display. Since carriage returns and line feeds are embedded in the PRINT statement, they need to be suppressed by a semi-colon at the end of the PRINT statement. OUTLCD may be used to execute specific functions.

SUMMARY OF OUTLCD COMMAND: OUTLCD writes a byte to the LCD command register (use DEVICE 6 and PRINT to write display data). If no display is present, OUTLCD may hang. OUTLCD can be used for many functions, summarized here. The principal uses are to set the character entry address (OUTLCD 128-255) and home the cursor (OUTLCD 2) while maintaining display contents.

Example: 10 OUTLCD 2

```

OUTLCD 1 ' Clear display and Data display RAM
OUTLCD 2 ' Display from first displayable character (home the cursor)
OUTLCD 64 - 127
    Selects Character Generator RAM and determines entry address. Subsequent
    PRINTs will write to the CG RAM. Refer to the manufacturer's data sheet..
OUTLCD 128 - 255
    Select Data Display RAM and determines entry address. Subsequent PRINTs will
    access this portion of the display. This is used to position the cursor.
        1-line display uses addresses 128 to 207
        2-line display uses addresses 128 to 167 (line 1) and 192 to 231 (line 2)
Other useful addresses for functions invoked with OUTLCD:
OUTLCD 16 Shift cursor left one position without data entry
OUTLCD 20 Shift cursor right one position without data entry
OUTLCD 24 Shift displayed characters left one position without data entry
OUTLCD 28 Shift displayed characters right one position without data entry

```

Caution should be exercise when using OUTLCD commands in the range of 3 to 63. INITLCD is the preferred command for functions in this address range. If the reset sequence is accidentally started, OUTLCD may hang microcontroller operation.

MULTITASKING EXAMPLE: As a final example, the following program combines the use of the keypad and an LCD Display in a multitasking routine: It displays the time-of-day on an LCD Display, updating every 100 milliseconds. Key strokes are simultaneously stored in a buffer. When the <Enter> key (lower right or "F" key) is pressed, the contents of the keypad buffer are displayed momentarily. The time-of-day display then resumes. Example [KEY1]

```

5  INTEGER X
10  STRING K$,KK$
15  START 1,10
16  START 2,5
17  INITLCD 0,2,4
18  REM Set the time and date here , if needed
20  STIME 12,34,56,1
22  SDATE 4,12,90

```

'Set the time to 12:34:56 pm
'Set the date to 4/12/90

"D key defined as <ESC> to delete entry
 "E key defined as backspace to delete entry
 "F key defined as <Enter>

